PTO/SB/33 (07-05)
Approved for use through xx/xx/200x. OMB 0651-00xx
U.S. Patent and Trademark Office: U.S. DEPARTMENT OF COMMERCE

PRE-APPEAL BRIEF REQUEST FOR REVIEW		Docket Number: 13361-071001
I hereby certify that this paper was filed with the Patent and Trademark Office using the EFS-WEB system on this date: February 1, 2008	Application Number 10/809,537	Filed March 24, 2004
	First Named Inventor Yosef Solt et al.	
	Art Unit 2189	Examiner Horace L. Flournoy

Applicant requests review of the final rejection in the above-identified application. No amendments are being filed with this request.

This request is being filed with a Notice of Appeal.

The review is requested for the reason(s) stated on the attached sheet(s).
Note: No more than five (5) pages may be provided.

I am the

☐ applicant/inventor.

☐ assignee of record of the entire interest.
See 37 CFR 3.71. Statement under 37 CFR 3.73(b) is enclosed. (Form PTO/SB/96)

☒ attorney or agent of record 52,713 (Reg. No.)

☐ attorney or agent acting under 37 CFR 1.34.
Registration number if acting under 37 CFR 1.34

/Alex Chan/
Signature

Alex Chan
Typed or printed name

(650) 839-5070
Telephone number

February 1, 2008
Date

NOTE: Signatures of all the inventors or assignees of record of the entire interest or their representative(s) are required. Submit multiple forms if more than one signature is required, see below*.

☒ Total of 1 forms are submitted.

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant	: Yosef Solt et al.	Art Unit	: 2189
Serial No.	: 10/809,537	Examiner	: Horace L. Flournoy
Filed	: March 24, 2004	Conf. No.	: 6150
Title	: BUFFER MANAGEMENT ARCHITECTURE		

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

PRE-APPEAL BRIEF REQUEST FOR REVIEW

This brief is filed in response to factual deficiencies in the Final Office Action mailed November 1, 2007.

Currently, claims 1-111 are pending in the action of which claims 9-17, 27-36, 46-55, 65-74, 84-93 and 102-110 have been withdrawn. Claims 1, 18, 56, 75 and 94 are independent, and claims 7-8, 24-25, 62-63, 81-82 and 100-101 have been indicated to contain allowable subject matter.

Section 102(e) Rejections

Claims 1-6, 18-23, 37-42, 45, 56-61, 64, 75-80, 83 and 94-99 are rejected as allegedly being unpatentable over **Shemla**. Applicants respectfully traverse this rejection.

Claim 1 recites in part writing **one of a plurality of sets** in an allocation memory into an allocation register, wherein the allocation memory includes a plurality of data elements arranged in the plurality of sets.

In the Office Action dated February 6, 2007 ("Office Action"), the Examiner asserted that Shemla's hash table 212 corresponds to the claimed allocation memory, and Shemla's hash table locations correspond to the claimed sets. *See*, page 4, lines 11-12 of Office Action.

In the Response filed August 21, 2007, Applicants argued, *inter alia*, that Shemla's hash table 212 does not include "a plurality of sets", because each of Shemla's hash table location is not an individual set or one that makes a plurality of sets collectively. *See*, page 30, last 2 lines

of the Response.

In the pending Final Office Action dated November 1, 2007 ("Final Office Action"), the Examiner maintains that Shemla's hash table locations are "sets", citing elements 212, 222, 224, 225 and 220 as the support. *See*, page 6, lines 16-18 of Final Office Action.

A. Shemla's Hash Table Are Not Sets

Applicants respectfully submit that Shemla's hash table locations are not sets. Rather, Shemla's hash table locations are source and destination addresses that have been transformed into table locations via hash functions (8:39-42). As described in Shemla, each hash table location stores its own associated address information (8:45-53). This allows Shemla to identify which locations are empty and which empty locations in the table are "next" for an incoming packet without using pointers (8:67-9:3). In one example, Shemla provides that a packet can jump from table location 6 to location 11 and to location 16 (or until which there's a match between the input address and the address associated with a location) (8:54-64). Shemla's hash table locations are not sets, but are address locations that indicate where a packet is to be transmitted. *See*, *also*, col. 9, lines 37-41.

B. Shemla Does Not Write The Alleged Sets In An Allocation Memory Into An Allocation Register

Further, Applicants respectfully remind the Examiner that claim 1 does not only recite a plurality of sets, but also recites **writing one of a plurality of sets in an allocation memory into an allocation register**. The Examiner indicates that Shemla's request register 64 is the claimed allocation register, thus suggesting that Shemla writes the hash table locations into the request register 64. *See*, page 3, lines 10-11 of Final Office Action.

Applicants respectfully submit that even assuming for the sake of argument (a point Applicants do not concede) that Shemla's hash table locations are sets, Shemla writes only buffer request messages into the request register 64. Nowhere does Shemla teach or suggest writing anything other than buffer request messages into the request register 64. As explicitly described in Shemla, the transfer manager 62, which is responsible for transferring a pending packet in the queue, writes a buffer request message into the request register 64 prior to transferring the packet

(5:37-43), which causes the empty list block 50 to allocate an available buffer in the DRAM 20 for the packet (5:44-47). There is nothing in Shemla that suggests that Shemla writes the hash table locations (or associated address information) into the request register 64.

C. Shemla Does Not Identify A Data Element In An Allocation Register Or Changing The Value Of The Data Element

Further, claim 1 recites in part **identifying a data element** in an allocation register having a value corresponding to an available buffer, and **changing the value** of the data element to a value corresponding to an allocated buffer.

In the previous Response, Applicants argued that Shemla does not identify a data element in the buffer request message. Rather, Shemla's single-bit data element, whose value can be changed based on the empty or full state of an associated buffer, resides in the empty list block 50 of the switching unit 34 (2:21-31; 6:37-45), not in the request transfer register 64.

In the pending Final Office Action, the Examiner maintains that Shemla teaches these features at col. 10, lines 15-22 and col. 5, line 61-col. 6, line 9. *See*, page 7, lines 1-2 of Final Office Action. Specifically, by indicating that Shemla's request register 64 is the claimed allocation register, the Examiner is suggesting that Shemla identifies a data element in the request register 64, and changes the value of this data element to a value corresponding to an allocated buffer.

Applicants respectfully reiterate that Shemla identifies an empty buffer and changes its value in the empty list block 50, not in the request register 64. Shemla's section at col. 10, lines 15-22, as discussed above, is directed to writing a buffer request message to the request register 64. This paragraph provides nothing about identifying a data element in the request register 64. The very fact that Shemla's empty list block 50 allocates an empty buffer 124 in the empty list 110 by changing the value of the empty buffer 124 (e.g., from "0" to indicate a buffer that is free to be written, to "1" to indicate a buffer that already stores a packet) (6:37-45) further evidences that Shemla: 1) identifies an available buffer in the empty list block 50, not in the request register 64; and 2) changes the buffer value of the available buffer in the empty list block 50, not in the request register 64. The relied upon portions of Shemla simply do not teach or suggest identifying a data element in an allocation register having a value corresponding to an available

buffer, and changing the value of the data element to a value corresponding to an allocated buffer, as recited in claim 1.

Applicants respectfully submit that reading Shemla's empty list block 50 as an allocation register also would not cure the deficiencies in Shemla. In such a scenario, Shemla also would fail to write one of a plurality of sets, for Shemla has provided no mechanism for writing the hash table locations into the empty list block 50.

For at least the foregoing reasons, Applicants respectfully submit that Shemla does not anticipate claim 1. Claims 2-8 depend from claim 1, and also are submitted to be allowable for at least the same reasons set forth above with respect to claim 1.

Claim 18

Claim 18 recites in part a buffer manager to **write one of a plurality of sets** into an allocation register, **identify a data element** in the allocation register, and **change the value of the data element**.

As discussed above, Shemla does not teach at least or suggest these features. For at least the reasons similar to those discussed with respect to claim 1, Applicants respectfully submit that Shemla also does not anticipate claim 18. Claims 19-26 depend from claim 18, and also are submitted to be allowable for at least the same reasons set forth above with respect to claim 18.

Claim 56

Claim 56 recites in part a buffer manager to **write one of a plurality of sets** into an allocation register, **identify a data element** in the allocation register, and **change the value of the data element**.

As discussed above, Shemla does not teach at least or suggest these features. For at least the reasons similar to those discussed with respect to claim 1, Applicants respectfully submit that Shemla also does not anticipate claim 56. Claims 57-64 depend from claim 56, and also are submitted to be allowable for at least the same reasons set forth above with respect to claim 56.

Claim 75

Claim 75 recites in part a buffer manager including means for **writing one of a plurality**

of sets into an allocation register, **identifying a data element** in the allocation register, and **changing the value of the data element**.

As discussed above, Shemla does not at least teach or suggest these features. For at least the reasons similar to those discussed with respect to claim 75, Applicants respectfully submit that Shemla also does not anticipate claim 75. Claims 76-83 depend from claim 75, and also are submitted to be allowable for at least the same reasons set forth above with respect to claim 75.

Claim 94

Claim 94 recites in part **writing one of a plurality of sets** into an allocation register, **identifying a data element** in the allocation register, and **changing the value of the data element**.

As discussed above, Shemla does not at least teach or suggest these features. For at least the reasons similar to those discussed with respect to claim 94, Applicants respectfully submit that Shemla also does not anticipate claim 94. Claims 95-101 depend from claim 94, and also are submitted to be allowable for at least the same reasons set forth above with respect to claim 94.

Please apply any other charges or credits to Deposit Account No. 06-1050.

Respectfully submitted,

Date: February 1, 2008

/Alex Chan/
Alex Chan
Reg. No. 52,713

Customer No.: 26200
Fish & Richardson P.C.
Telephone: (650) 839-5070
Facsimile: (650) 839-5071